PATENT ABSTRACTS OF JAPAN

(11)Publication number : 11-184744
(43)Date of publication of application : 09.07.1999

(51)Int.Cl. G06F 12/00
G06F 12/00
G06F 13/00

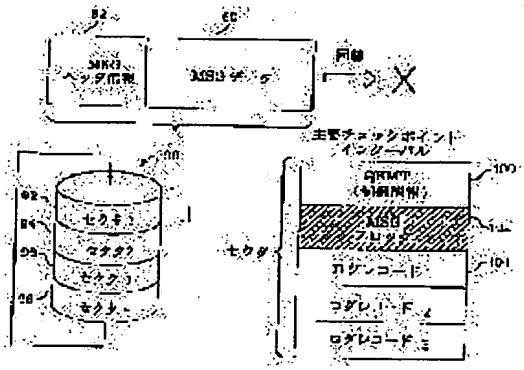
(21)Application number : 10-247027
(22)Date of filing : 01.09.1998

(30)Priority
Priority number : 97 963188 Priority date : 03.11.1997 Priority country : US

(71)Applicant : MITSUBISHI ELECTRIC INF TECHNOL CENTER AMERICA INC
(72)Inventor : WONG DAVID W H
SCHWENKE DEREK L

(54) MESSAGE QUEUING SYSTEM

(57)Abstract:
PROBLEM TO BE SOLVED: To enable speedy recovery from a server defect by preserving and storing a message and its condition in an efficient single file on a single disk.
SOLUTION: Message data 60 and message header information 62 are stored in continuous sectors 92, 94, 96 and 98 of a single disk storage device 90. Then, the message and header information can be stored in accessible order by utilizing a queue entry management table. The message data are not stored in all the sectors 92, 94, 96 and 98 but are continuously stored. In order to enable access to data stored in this file 90, the queue entry management table is provided with the input of control information and sector information containing a message block 102 and a log record 104. All the data are designed so as to peculiarly specify the sector where related data and header can be found out.



LEGAL STATUS

[Date of request for examination]

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

(19)日本国特許庁（ＪＰ）　(12)公開特許公報（Ａ）　(11)特許出願公開番号

特開平11－184744

(43)公開日　平成11年(1999)７月９日

(51)Int.Cl.[6]	識別記号	FI	
Ｇ０６Ｆ　12/00	５３１	Ｇ０６Ｆ　12/00	５３１Ｒ
	５１８		５１８Ａ
13/00	３５１	13/00	３５１Ｎ

審査請求　未請求　請求項の数７　ＯＬ　（全 14 頁）

(21)出願番号　特願平10－247027

(22)出願日　平成10年(1998)９月１日

(31)優先権主張番号　０８／９６３１８８

(32)優先日　1997年11月３日

(33)優先権主張国　米国（ＵＳ）

(71)出願人　597067574
ミツビシ・エレクトリック・インフォメイション・テクノロジー・センター・アメリカ・インコーポレイテッド
ＭＩＴＳＵＢＩＳＨＩ　ＥＬＥＣＴＲＩＣ　ＩＮＦＯＲＭＡＴＩＯＮ　ＴＥＣＨＮＯＬＯＧＹ　ＣＥＮＴＥＲ　ＡＭＥＲＩＣＡ，　ＩＮＣ．
アメリカ合衆国、マサチューセッツ州、ケンブリッジ、ブロードウエイ　201

(74)代理人　弁理士　曾我　道照　（外６名）

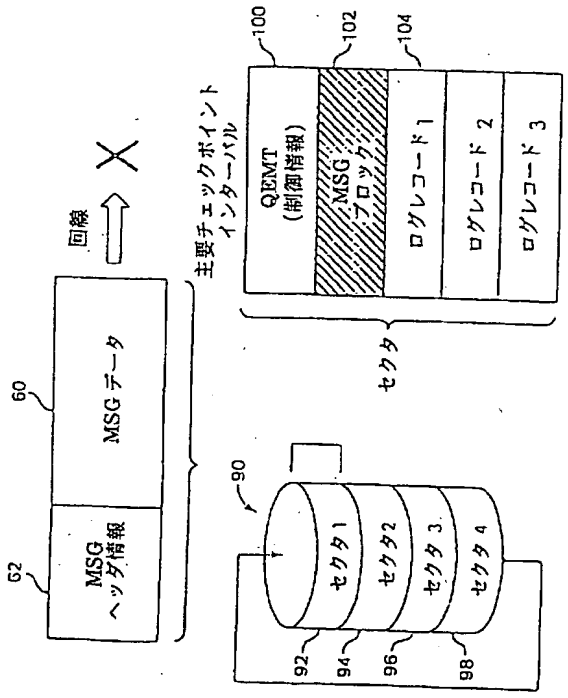
最終頁に続く

(54)【発明の名称】　メッセージキューイングシステム

(57)【要約】

【課題】　サーバ故障から迅速に回復できるようにメッセージおよびその状況を単一ディスク上の効率的な単一ファイルに保存し格納するメッセージキューイングシステムを提供する。

【解決手段】　メッセージおよびその状況を格納する単一ディスク単一ファイル格納システムは、データディスク、インデックス構造ディスクおよびログディスクへの書込みを消去する。単一ディスク単一ファイル格納は、同一ディスク上の隣接スペースにおいてすべての情報を集束させることによって可能となる。集束された情報を追従するために使用される固有のキューエントリマップテーブルは、制御情報と、メッセージブロックと、ログレコードとを含み、同時に新規のレコードを書込む際保存されたデータをトラバースするために書込みヘッドがバックアップする必要の全くない単一ファイルディスク格納装置を含む。



1

【特許請求の範囲】

【請求項1】　キューの状態と、メッセージキューデータと、ログレコードを含んで連結されたメッセージキューを有するトランザクションメッセージを伝送する手段と、

受容サイトで、上記メッセージキューデータと上記ログレコードとが組み合わされたオンディスクファイル構造を用いている単一ディスクの単一ファイルに上記トランザクションメッセージキューデータを格納する手段とを含むメッセージキューイングシステム。

【請求項2】　上記単一ディスクにアクセスする読出し／書込みヘッドと、書込み動作中に上記ヘッドを単一フォワード方向で駆動する手段とをさらに含むことを特徴とする請求項1に記載のメッセージキューイングシステム。

【請求項3】　上記ディスクの予め選択された場所に配置されて、制御情報ブロックと、少なくとも１つのメッセージブロックと、少なくとも１つのログレコードとを有するキューエントリマネジメントテーブルをさらに含む請求項1記載のメッセージキューイングシステム。

【請求項4】　上記予め選択された場所が上記ファイルの開始から固定されたオフセットと対応し、これによってメッセージキューデータの最新状況を迅速に識別することができる請求項3記載のメッセージキューイングシステム。

【請求項5】　上記受容サイトで、上記情報の前の最新キューエントリマネジメントテーブルに呼応する上記伝送の割込み時に、上記メッセージキューを回復する手段をさらに含み、これによって、上記最新キューエントリマネジメントテーブルに含まれる情報から受信され格納された最新有効情報を見つける請求項4記載のメッセージキューイングシステム。

【請求項6】　上記ファイルが複数のセクタに分割され、上記テーブルが有効情報を有するセクタの位置に対するチェックポイントを構成するように上記オフセットがキューエントリマネジメントテーブルをセクタの最初に配置し、これによって上記割込み前の最新の有効情報が、上記最新のテーブルを包むセクタの識別を通して迅速に見つけ出される請求項5記載のメッセージキューイングシステム。

【請求項7】　上記マネジメントキューテーブルが隣接ブロックに書込まれ、単一ファイルの結果、フォワード書込み方向および隣接メッセージキューデータブロックシステムによりシーク時間を最小限に抑え、伝送割込みから完全かつ迅速な回復を増大させる請求項2記載のメッセージキューイングシステム。

【発明の詳細な説明】

【０００１】

【発明の属する技術分野】この発明は、メッセージキューイングに関し、より詳しくはクライアントサーバおよびモバイルエージェントアプリケーションのための迅速で信頼性のあるトランザクションメッセージキューイングシステムに関し、さらに、当該システムのためのログベースデータアーキテクチャに関する。

2

【０００２】

【従来の技術】メッセージキューイングは、その本来の同期処理および非同期処理をいずれも可能にするという柔軟性により、それぞれ異なるコンピュータシステム上のアプリケーション間で最も基本となる通信範例である。メッセージキューイングミドルウェア基幹は、一般的なクライアントサーバならびにモバイルエージェント計算処理、すなわちワークフロー計算処理、オブジェクトメッセージ伝送、トランザクションメッセージ伝送およびデータ複写サービスのいずれにおいても膨大なアプリケーションドメインについて非常に柔軟性のある骨組みである。

【０００３】

【発明が解決しようとする課題】トランザクションメッセージ伝送の場面ではデータが転送中に紛失することがたびたびある。金融業界では、ある場所から別の場所に転送される銀行の取引記録が、サーバ不良、送信回線の不具合またはその他人為的なものによって紛失する可能性があるが、これが金融業界で起これば大惨事になる。エラーが発生したことを迅速に突き止めることができ、かつデータが有効であった既知のポイントからデータの再構築を可能にすることがシステムマネージャに課せられる。

【０００４】クラッシュ前の当該システムの最新状況を再構築するために当該システムがいわゆるログファイル全体を走査することによって、過去においてエラーが発生した地点を確定する。関連するタイムスタンプを有するログファイルを常時利用して、メッセージおよびログファイルに含まれるデータを識別する。しかしながら、最新状況を確認するためにログファイル全体を走査するには、１，０００ものログレコードを走査する必要がある。

【０００５】エラーの発生した地点を突き止め、かつその発生地点からファイルを再構築するログレコード全体をそうさすることは、非能率な方法というだけでなく、従来のシステムでは、２種類のディスクファイルが必要であった。このうち、ひとつはデータファイルとして、他方はログファイルとして機能する。さらに、ログエントリとデータファイルまたはセクタの間の相互関係は、従来のセクタが識別性のない順序で格納され、ログファイルとセクタとの間のマッピングが所要時間の多少かかるプロセスであるたため複雑である。

【０００６】さらに技術背景では、一つのポイントから別のポイントに伝送されるデータレコードが損なわれることのないような格納装置を提供可能にするようなメッセージキューイングが一般的に使用されることは理解さ

れよう。たとえば、ある場所でエラーが発生しデータが紛失した場合でも、メッセージキューイング本来の格納装置が機能してデータを第2の場所で再構築可能である。

【０００７】例として、特に株取引では、取引中の割込みは数時間というよりは数分に極限されることが望ましい。しかし、時に、システムサーバが停止した場合、その時のシステムでの取引数により、回復に2時間から8時間もかかることがある。したがって、停止時間や破損ファイルの見つけ出しと再構築に所要する時間と費用を最小限に抑える必要がある。

【０００８】なお、ここで用いられるキューファイルは、伝送中のメッセージの物理的格納装置を表す。キューファイルは、未完了操作のための保持用セルと称することもできる。すなわち、基本的には、所定のメッセージを受信する受け手がそこにいない場合、メッセージをキューファイルに保持し、後で送出できるようにすることを意味する。したがって、キューファイルは送信された情報の保持に信頼性を与える。

【０００９】さらに、従来のシステムにおいて、回復データは、キューファイルそれ自体により提供されるものではない。したがって、エラーまたはデータの紛失が発生した場合、キューファイルはファイル状態を確認するために利用されていない。すなわち、以前に不正処理されていないデータからデータファイルを再構築するためにキューファイルは使用されていない。従来のシステムでは、キューファイルそれ自体により回復データが提供されることはない。

【００１０】メッセージキューイングの実世界のアプリケーションへの応用例についての別の例では、メッセージキューイング基幹のモバイルエージェントを用いたリアルタイムオンライントランザクション処理の支援の仕方に絡んでいる。本例では、顧客はたとえば地理的に分散した支店を有する銀行である。顧客の口座が作られ、その口座が開設された地方支店で保管される。例示の目的のために、これを口座の本拠地支店（ home branch）と称する。各口座の写しが本店においても保管される。口座の読出し取り操作を地方支店または本店のいずれからも行うことが可能である。しかしながら、本拠地支店にある写しと本店の写しとが同等に更新することが要求される。

【００１１】更新の要求が本拠地支店で発生すると、地方支店にあるコピーを更新しなければならない。この更新によって、次にキューに加える（ enqueue）要求をキューマネージャまたはキューサーバに自動的に送出するエージェントを始動することが可能である。このキューマネージャは、広域ネットワークを介して別のキューマネージャに対しての要求をキューから外し（ dequeue）、このキューマネージャが、今度は、ミラーオフィスにある口座のデータベースサーバに対しての更新要求

をキューから外す。

【００１２】メッセージキューは、本例では非同期の信頼性のある処理を提供する。非同期処理は、ある位置でのデータベースの更新によって起動するエージェントから始まる。エージェントは、更新要求をメッセージキューマネージャに対して非同期で送出するが、応答を待つ必要はない。メッセージキューマネージャは、要求者が応答を待つ必要なく処理を継続できるように要求についての保持セルとして機能する。さらにメッセージキューマネージャは、本例では、更新の要求の受け手が、トランザクションメッセージキューとして当該業界において知られている二相コミットプロトコル（ Two Phase Commit Protocol）と呼ばれる周知のハンドシェークプロトコルを介して受信状況をひとつひとつ確認するまで、更新要求のコピーをキューで保持することで信頼性を提供する。

【００１３】これらのタイプのメッセージキューイングシステムは、これまで信頼可能に動作したが、メッセージキューに添付されるメッセージを格納するために別個のキューデータおよびログレコードファイルを使用するデータ構造に依存するものであった。このような構造は、サーバのクラッシュ時における迅速な修復を妨げ、2種類の格納用ディスクを必要とする。一つはデータのためのディスクであり、他方はログレコードのディスクである。さらに、従来のメッセージキューイング構造は、通常、効率よく作業するための予備のハードウェアを必要とせずには書込み動作が最適化されることはない。また、メッセージ滞留時間の短い高性能のスループットシステムには適切ではない。上述の別個のキューデータおよびログファイルにもまた、非信頼性が必要以上のレベルで取り入れられている。これは、ファイル不正処理および媒体不良の二点が潜在的に含まれているからである。さらに、メッセージキューイングシステムの業務管理担当者のために最初から回復に要する仕事量が予め定められる手段は通常存在しない。

【００１４】なお、上述のシステムは、Digital Equipment Corporation社のDECmessageQ、IBM社のMQシリーズおよびTransarc社のEncina RQSとして市販化されている。

【００１５】この発明は上述した点に鑑みてなされたもので、メッセージおよびその状況を単一ディスク上の効率的な単一ファイルに保存し格納することによってサーバ不良から迅速に回復することが可能なメッセージキューイングシステムを提供することを目的とする。

【００１６】

【課題を解決するための手段】この発明に係るメッセージキューイングシステムは、キューの状態と、メッセージキューデータと、ログレコードを含んで連結されたメッセージキューを有するトランザクションメッセージを伝送する手段と、受容サイトで、上記メッセージキュー

5

データと上記ログレコードとが組み合わされたオンディスクファイル構造を用いている単一ディスクの単一ファイルに上記トランザクションメッセージキューデータを格納する手段と含むものである。

【００１７】また、上記単一ディスクにアクセスする読出し／書込みヘッドと、書込み動作中に上記ヘッドを単一フォワード方向で駆動する手段とをさらに含むものである。

【００１８】また、上記ディスクの予め選択された場所に配置されて、制御情報ブロックと、少なくとも1つのメッセージブロックと、少なくとも1つのログレコードとを有するキューエントリマネジメントテーブルをさらに含むものである。

【００１９】また、上記予め選択された場所が上記ファイルの開始から固定されたオフセットと対応し、これによってメッセージキューデータの最新状況を迅速に識別することができるものである。

【００２０】また、上記受容サイトで、上記情報の前の最新キューエントリマネジメントテーブルに呼応する上記伝送の割込み時に、上記メッセージキューを回復する手段をさらに含み、これによって、上記最新キューエントリマネジメントテーブルに含まれる情報から受信され格納された最新有効情報を見つけるものである。

【００２１】また、上記ファイルが複数のセクタに分割され、上記テーブルが有効情報を有するセクタの位置に対するチェックポイントを構成するように上記オフセットがキューエントリマネジメントテーブルをセクタの最初に配置し、これによって上記割込み前の最新の有効情報が、上記最新のテーブルを包むセクタの識別を通して迅速に見つけ出されるものである。

【００２２】さらに、上記マネジメントキューテーブルが隣接ブロックに書込まれ、単一ファイルの結果、フォワード書込み方向および隣接メッセージキューデータブロックシステムによりシーク時間を最小限に抑え、伝送割込みから完全かつ迅速な回復を増大させるものである。

【００２３】

【発明の実施の形態】この発明において提供されるメッセージキューイングシステムは、従来のメッセージキューイングが有する課題を解決するために、メッセージおよびその状況を単一ディスク上の効率的な単一ファイルに保存し格納することによって、サーバ不良からの回復を迅速に行うことが可能となる。メッセージおよびその状況を格納する単一ディスク単一ファイル格納装置システムによって、3種類のそれぞれ異なるディスク、すなわちデータディスク、インデックス構造ディスクおよびログディスクへの書込みが消去される。単一ディスク単一ファイル格納装置は、同一ディスク上の隣接スペースにおいてすべての情報を集束させることによって可能となる。

6

【００２４】この結果、すべての書込みは、書込みヘッドの一つの掃引動作に含まれ、書込みヘッドは一方向にのみ一度だけ移動して、メッセージの書込み開始を必要とし、その状態を格納する領域を見つけ出す。集束された情報を追従するために使用される固有のキューエントリマップテーブル(Queue Entry Map Table) は、制御情報と、メッセージブロックと、ログレコードと、新規のレコードを書込む際に保存されたデータをトラバースするために書込みヘッドがバックアップする必要が全くない単一ファイルディスク格納装置とを同時に含む。さらに当該システムは、ログファイル全体を走査する必要なく破損ファイルを見つけることができる。

【００２５】最新の有効データを見つけるために、制御チェックポイント間隔システムを利用して最新の不正処理されていないデータを見つけることができる。走査を行い、最新のチェックポイント間隔を見つけることによって、最後のキューをすぐに認識できる。チェックポイント後にログレコードの走査を行い、すべてのメッセージの最新状況を設定する。上述のシステムによって従前のシステムよりも少ない時間の位数でデータ回復を行うことができる。同時に、効率的なフォワード方向への書込み方法を確立させることによって、順序づけられていないセクタを介して検索する必要がなくなる。

【００２６】一実施態様によると、最後尾のセクタに新規のレコードを追加することによって先行のセクタを更新し、ファイル状態が変更されたことを示す巡回循環バッファ用システム (circular wrap around buffering system)を用いることによって、開放され、有効メッセージおよび／またはログレコードをもはや保持していない先行のブロックを再利用する。

【００２７】したがって、この発明は、トランザクションメッセージキューイングシステムのためのログベースデータ構造(アーキテクチャ) を提供するものであり、当該システムは、メッセージキューデータおよびログレコードの組合わせオンディスクファイル構造を利用するものである。この発明の一実施態様によると、単一ディスクのキューデータ／ログレコード組合わせファイルでは、書込み動作の性能および信頼性が向上し、同時に使用ディスク数が減少する。

【００２８】上述のように、システムクラッシュの回復は、エラーの発生場所を突き止める際にすべてのログレコードを通して検索する必要のないキューエントリマップテーブルを使用することによって加速される。さらに、キューエントリマップテーブルを使用することによって、システム業務管理担当者に対して拡張性および柔軟性をもたらすキューデータファイルに要件の数を当初から割り当てることが可能である。

【００２９】さらに上述したように、当該システムは、格納装置の再利用のためにキューデータファイルの循環(ラップアラウンド) が潜在的に存在することを暗示す

る巡回キューを利用するものである。これによって、キューが循環する（ラップアラウンド）場合、まだ有効かもしれないキューデータおよび／またはログレコードが次の書込み動作によって確実に上書きされないように予約表（リザベーションテーブル）または自由空間（フリースペース）ヒープを維持することを要求される。

【００３０】一実施態様によると、キューデータ格納装置構造（アーキテクチャ）は、キューの固定サイズに基づいてキューマネージャを最初に初期化する場合に作成される単一フラットファイルからなる。初期キューの作成は、メッセージキューイングシステムにおけるピーク負荷、たとえば時に所定の時点でメッセージキューに予想される入力の最大数についてのシステム業務管理担当者の感覚に基づいて行われる。キューデータファイルにおける各メッセージは、メッセージヘッダおよびメッセージ本文を含む。メッセージ本文は、メッセージの内容を含み、メッセージヘッダに続く次の隣接ブロックのディスクに格納される。

【００３１】上記実施態様では、キューデータファイルは、実行時に拡張可能な所定数の論理セグメントまたはセクタに分割される。各セグメントは、キューエントリマップテーブル（QEMT）のコピーを包含し、各セグメントの冒頭にこれが格納される。QEMTは、キューファイル全体に格納されたキューエントリおよびログレコード情報についての制御情報を包含する。メッセージヘッダ、メッセージ本文およびログレコードは、QEMTの後にメッセージデータおよびログレコードブロックの潜在的な混合と共に格納される。

【００３２】理解されるように、QEMTのサイズは、ユーザがキュー作成時に定義するキューエントリの予測最大数に依存する。ログレコードは、決定論的バイト数を取るため、キューデータファイルは、ログレコードと、メッセージヘッダと、メッセージ本文と、QEMTの混合のデータタイプから構成される。新規セグメントがキューデータファイルに到達すると、その新規セグメントの冒頭に新規QEMテーブルがディスクに書込まれ、メッセージおよびログレコードがQEMテーブルに続く。最も小さいオンディスクデータのタイプはログレコードであるため、１ブロックがログレコードのサイズである場合、キューデータファイルのセグメントは、ブロックからなるように定義される。このように実施性を高めることは、検索アルゴリズムの開発を容易にする。

【００３３】トランザクションメッセージキューイングシステムの状態は、QEMTに包含される制御情報によって捕獲される。QEMTは、各自コピーを維持する各スレッドよりむしろ多重スレッドを作動させることが可能な静的データ構造として定義される。

【００３４】ログベースデータ構造（アーキテクチャ）の結果、当該発明は、既存のトランザクションメッセージキューイングデータ構造（アーキテクチャ）において

多数の改良を提供するものである。書込み動作の性能が既存のメッセージキューイング構造（アーキテクチャ）において改良され、この発明をベースにしたメッセージキューイングシステムによって、高速化の銀行処理アプリケーション等メッセージ滞留時間を短縮した高性能化スループットシステムが確実に達成される。さらに当該システムは、様々な帯域幅を有するネットワークおよび／または信頼性のないネットワークを介してエージェントの搬送の際に根底をなす信頼性のあるメッセージ送信基幹（インフラストラクチャ）についても適用可能である。

【００３５】さらに、メッセージデータおよびログレコードの書込み動作は、常にフォワード方向に行われ、これらはいずれも同一のディスクファイルに格納されることができる。

【００３６】また、本システムは、トランザクションメッセージキューイングシステムの信頼性を向上させるものである。当該ログベースデータ構造（アーキテクチャ）において、ファイルの不正処理が分割キューデータおよびログレコードファイルとの２つの潜在的なファイル不正処理の場面に対して起こり得る一つの場所が存在する。また、使用されるディスクファイルが少なくなるため、信頼性も高まる。キューデータ／ログレコードの組合わせファイルは、公知のACID特性の原始性（Atomicity）、一貫性（Consistency）および隔離性（Isolation）のそれぞれの特性に忠実である。さらに、明らかなように、既存のRAID技術を駆使して透過性のある二重書込みを行うことが可能である。

【００３７】当該システムにおいて、これによって得られたメッセージキューイングシステムによって、先入れ先出し（First In First Out）、後入れ先出し（Last In First Out）または優先順位ベースのメッセージデータアクセスを含むいずれのメッセージデータアクセス方法も支援可能となる。また、同時にシステムクラッシュからの回復に要する時間を短縮することができる。従来のアプローチではログレコードのファイル全体の全データを走査するが、当該システムでは、最新のチェックポイントを決定するために一部のキューエントリマップテーブルをまずテストすることのみ要求される。そして次にそのセグメント内にあるログレコードに走査を進める。

【００３８】さらに、この発明によって、業務管理担当者は、キューデータファイルのセグメント数、続いてチェックポイント間隔の数を最初から予め定めることによって、システム回復に要する仕事量を調整できるため、当該システムは、メッセージキューイングシステムの業務管理に対して拡張性および柔軟性を提供する。したがって、システム業務管理担当者は、チェックポイントの書込みの合計金額を先に支払うため、回復時に拡張ログレコードの走査を行う場合の高額の支払いを防ぐ。この

9

トレードオフ(交換条件) を調整および微調整することによって、アプリケーションの要件およびドメインを適合させることができる。

【００３９】上記の利点は、キュー制御情報と、メッセージデータと、メッセージ動作のトランザクションログレコードとを包含する予め割り当てられたオンディスクキューバッファを使用することによる。オンディスクキューバッファは、多数のセグメントまたはセクタから構成される。各セグメントは、同一の所定のブロック数から構成される。各セグメントの冒頭には上述のキューエントリマップテーブルがあり、個別のキューエントリの状況に関する制御情報データと、ディスク上にありメッセージが物理的に格納されるポインタオフセットとを包含する。キューエントリマップテーブルは、メッセージキューイングシステム全体についての固定のチェックポイント間隔として機能する。メッセージ動作のメッセージおよびトランザクションログレコードは、メッセージブロックとログレコードブロックが組み合わせられるようなセグメント内のブロックに格納される。また、ある特定のメッセージのログレコードを当該メッセージに隣接するように格納することは要求されない。

【００４０】当該発明の特徴として、ディスクヘッドに対して常にフォワード方向になるようにメッセージデータ書込み操作を行う。また、ポインタがトラバースする必要なく連続してメッセージをディスクに格納する。さらに、ログレコード書込み動作は、常にディスクヘッドに対してフォワード方向になるように行う。ログレコードは、二相コミットプロトコルに基づいてメッセージ操作の状態の変化が書込まれる。したがって、ログレコードは、準備(Prepare)、準備完了(Prepared)、コミット(Commit)、アボート(Abort)そして確認(Acknowledge)の各メッセージが遠隔のキューマネージャから書込まれることができる。

【００４１】別の固有の特徴として、キュー全体がシングルパスで走査されることができる。さらに、オンディスクの不要データの集積は常に線形プロセスである。さらに、多数のキューエントリマップテーブルが、キューマネージャの適切なシャットダウン時にディスクに格納される最新のテーブルの固有のシーケンス番号と共に、同一ファイルに存在している。

【００４２】重要なこととして、読出し動作は、先出し先入れ、後入れ先出しまたは優先順位ベースの方策に従い、これらの方策のいずれかを実行するために特別な規定を必要としない。

【００４３】さらに、回復の手順は、キューエントリマップテーブルのタイムスタンプのみを検索することによって加速される。これは、最新のキューエントリマップテーブルが回復プロセスの開始状態として機能するからである。当該テーブルに続くログレコードは順に読み出され、最後の既知のチェックポイント後に行われる変更

10

に反映ために、この最新のキューエントリマップテーブルのインメモリコピーが変更される。

【００４４】次に、具体的な実施の形態について説明する。まず、図1を参照すると、更新された口座情報を支店から本店へ伝送する目的のために、メッセージキューイングシステム１０が銀行の支店１２と本店１４との間に設置される。また、銀行の異なる支店のそれぞれの端末機１６、１８および２０にデータがそれぞれ入力される。このデータは、各支店のローカルデータベースサーバ２２、２４および２６にそれぞれ格納され、各データベースサーバは独自のローカル格納装置を有し、ここでは参照符号２８によって示される。

【００４５】データベースサーバの出力は、一連のメッセージキューイングサーバ３０、３２および３４にそれぞれ接続され、それぞれが独自の格納装置を保有し、ここでは参照符号３６で付されている。

【００４６】メッセージキューイングサーバは、広域ネットワーク４０に対してその出力を行う。該ネットワークは、この出力を本店にあるメッセージキューイングサーバ４２に接続し、このサーバは、図示のように、各格納装置４４をそれぞれ有している。メッセージキューイングサーバ３０、３２および３４は、図示のように、接続された格納装置５２を有するデータベースサーバ５０と広域に通信を行う。メッセージキューイングサーバ４２の出力は、図示のように、接続された格納装置５２を備えるデータベースサーバ５０と接続される。このデータベースの情報は、本店の端末機５４において閲覧が可能である。

【００４７】メッセージキューイングシステムの目的は、更新された口座情報を本店に備えるために支店から信頼して伝送することを可能にすることである。また、本社と直接接続しているかどうかに関係なく、支店におけるトランザクションが進行できることも重要である。

【００４８】次に、図2を参照すると、従来、６０および６２で図示されるようなメッセージおよびヘッダは、データディスク６４のセクタ６６、６８、７０および７２に格納され、このセクタにおいてメッセージおよび随伴するヘッダがランダムに配置されていた。

【００４９】同時に、メッセージ状態の情報は、上記データディスクに格納されている各メッセージについてのレコードを含む、ログディスク８０に格納され、着信順位およびデータディスクにおける位置を含んでいた。さらに、トランザクションの状況は、メッセージおよび対応するヘッダそれぞれについてログディスク８０にログインされていた。

【００５０】「Ｘ」８２で示されるように、伝送が割り込まれる場合、従来、ここでは８４で図示されるように、伝送割込み直前のデータディスクファイルの最新状態が再構築できるようにログファイル全体を走査することが要求されていた。前述したように、これは時間のか

11

かるプロセスであり、クラッシュする直前のシステムの状態を再構築するためには、ログファイル全体が走査されなければならない。メッセージおよびヘッダの情報がデータディスクの非連続セクタに格納されるため、状況ははるかに複雑になる。また、伝送割込み時に不正処理されないメッセージを見つけるためにログファイルとデータファイルとの相互通信が要求される。

【００５１】次に、図3を参照すると、当該システムにおいてメッセージデータ６０およびメッセージヘッダ情報６２は、単一ディスク格納装置９０の、ここでは９２、９４、９６および９８で図示される連続するセクタに格納される。キューエントリマネジメントテーブルを利用することによってアクセス可能な順番にメッセージおよびヘッダの情報が格納されることはこの発明の特徴であり、後述するチェックポイントシステムを介してメッセージデータを見つける。

【００５２】メッセージデータがセクタすべてに格納されず、むしろ上述のように連続して格納されることは明らかである。ファイル９０に格納されたデータにアクセス可能とするために、キューエントリマネジメントテーブル（すなわちQEMT）は、制御情報１００の入力と、メッセージブロック１０２とログレコード１０４とを含むセクタ情報を含む。これらはすべて関連データおよびヘッダが見つけられるセクタを固有に特定するように設計される。したがって、QEMTは、このようにしてシステム状態を特定する。

【００５３】図4、図5および図6と関連して明らかなように、キューエントリマネジメントテーブルは、メッセージデータとヘッダ情報との間に分散されたファイル９０に格納される。

【００５４】次に、図4を参照する。一実施態様では、ファイル９０は、隣接するセクタがここでは１０６で図示される情報ブロックを有するように構成され、情報ブロックは、矢印１０８で図示されるように左から入り、左から入るブロック番号1および右から出るブロック番号１３によって図示されるようにファイルを左から右へトラバースする。隣接するブロックおよびファイルを通してのフローは、変化しないいわゆる書込み方向を作成することが理解されよう。

【００５５】次に、図5を参照する。上述のQEMT制御情報ブロック１００は、QEMT制御情報ブロック１００の位置がファイル９０を通して周知のオフセットでのチェックポイントを特定するように、その他の隣接ブロック１０６間に分散される。

【００５６】QEMT制御ブロックを一定の間隔で分散する目的は、場合によって、特定のメッセージデータおよびヘッダ情報を含む完全なシステム状態を、チェックポイント番号またはチェックポイント間隔を単に指定することによって迅速に見つけることである。その結果、有効情報を有する最後であるとしてチェックポイント間隔が

12

識別されると、QEMTブロックがどこで有効データが見つけられるか、ならびにその身元および位置を指定した後に隣接ブロックが書き込まれるように、制御QEMT制御ブロックのいずれかの側にメッセージデータおよびログレコードブロックを有することができる。

【００５７】別の説明として、QEMT制御ブロックは、周知の位置を回復プロセスに提供して当該システムの状態を調査する。

【００５８】次に、図6を参照すると、ブロック１０６は１１０で図示されるように、メッセージデータブロックとして、または１１２で図示されるように、増分ログブロックとしてブロック１１２を図3中のログレコード１０４に対応させながら利用される。これらのログレコードは、隣接する下流ブロックでのメッセージへの状態変化を記録する。なお、制御ブロックは、ファイルの調査の開始に一部の既知のポイントのみ与え、一方ログレコードはファイルにおける個別のメッセージに関する情報を提供する。

【００５９】図3に戻って参照すると、ログレコード１０４は、開始点がＱＥＭＴ制御ブロックで表されるデータに関連する多数の連続ログレコードの一つでしかないことが理解されよう。これらのログレコードは、先行のメッセージブロックにおける情報への変更を、その特定のメッセージブロックへの変更についての経過を完全に付するように、記録する。

【００６０】再び図6に戻ると、所定の数のメッセージブロックは、チェックポイント後に生じた追加メッセージデータブロックを特定するQEMT制御ブロックによって境界が示されることに留意する。このセクタ内にトランザクションログレコード１１２がある。ログレコードT_1は、メッセージブロックのいずれか一つにおいての変更を記載できることが明らかである。矢印１１４からわかるように、情報は左から右へ流れる。この場合、トランザクションログレコードT_1は、当該システムにおけるいずれのメッセージについても状態の変化を記述するが、これは、メッセージは受信されてもはや保持する必要がないという確認、またはメッセージは送信されたがまだ受信されていない、または確認されていないという確認の場合もある。さらに、上記は、この種のシステムにおいてメッセージを確実に伝送するために2パスハンドシェーキング技術（two pass handshaking technique）を反映している。

【００６１】たとえば、トランザクションログレコードT_1は、新規メッセージがその特定のポイントでファイルに追加されたことを示してもよい。ログレコードの作成時にログレコードの位置が書込みヘッドによって決定されることは理解されよう。そこで、ログレコードが時間T_1で作成される際、書込みヘッドはファイル中のある特定のポイントに存在するが、ログレコードは、全ファイル構造の中のいずれの位置においてもトランザ

13

クションおよびメッセージ照会することができる。

【００６２】同様に、トランザクションログレコードＴ２、Ｔ３およびＴ４　は、これらのログレコードを時に連続して投稿しながら、これらのメッセージが状態を変えたことを反映する。

【００６３】QEMIブロックおよびログレコードブロックは、単一ファイル構造に挿入可能であり、さらに単一ファイル構造は、一実施態様において一方向に情報の流れを有するため、先行技術の２ファイル構造を完全に排除することが可能である。さらに、QEMIブロックおよびトランザクションログレコードブロックを利用することによって、不正処理されないこれらのメッセージを固有に特定して情報割込みの影響を早期診断でき、システム故障後のシステム状態を素早く回復できる。

【００６４】次に図7を参照すると、キューエントリマネジメントテーブルのヘッダの構成が１２０に図示されている。これによって明らかであるように、一実施態様によると、ヘッダは、キューファイルのセグメント数１２２と、セグメントサイズ１２４と、QEMIシーケンス番号もしくはタイムスタンプ１２６と、前回のセグメントにおける最後尾のログレコードのシーケンス番号１２８と、現行セグメント数１３０と、キューヘッドポインタ１３２と、キューテイルポインタ１３４と、現行のセグメントにおいて次に利用可能なブロック１３６と、QEMI入力の一覧１３８と、ディスクブロックの予約表１４０と、調整者（コーディネータ）として動作する係属中のトランザクション一覧１４２、および受け手として動作する係属中のトランザクション一覧１４４とを含む。ヘッダに含まれる情報は、回復プロセスの支援情報であることが理解されよう。

【００６５】次に図8を参照すると、QEMI入力１３８は、それぞれシーケンス番号１４６と、メッセージＩＤ１４８と、Q_{put}またはQ_{get}のいずれかであるメッセージ動作モード１５０と、メッセージ受け手のノード名１５２と、メッセージ受け手のサーバ名１５４と、「アクティブ」、「ペンディング」、「アボート」または「コミット」のいずれかであるトランザクション状況１５６と、受信者によって受信された最後の既知の応答である参加者２ＰＣ投票(vote)１５８と、一組の追加フラグ１６０と、メッセージのポインタオンディスク位置１６２とを含む。したがって、キューエントリマネジメントテーブルは、ファイルの状況に関する正確な情報を提供し、より詳しくは任意のキューエントリに関する情報を提供する。

【００６６】次に図9を参照すると、ここでは、単一メッセージは隣接ブロックに格納されるため、再処理は、隣接ブロックを読み出し返す（read back）ことを含む。この結果、読出し操作中のヘッドの動作を軽減する。

【００６７】要するに、先行技術では、読出しを実行するために隣接しないブロックを読出しヘッドがトラバースすることを要求し、したがって、相当時間を要する可能性があった。当該システムにおいては、メッセージは隣接ブロックに格納されているため、読出し操作時にこれらの隣接ブロックをトラバースすることのみ必要となる。同様に、続いて起こる書込み操作ではヘッドは限定されたファイル量のみトラバースする。

14

【００６８】要するに、次の書込みについてフォワード方向の流れがあり、循環するため、データは隣接ブロックに構成され、ここから上記の利点が生じる。

【００６９】次に図１０を参照すると、図6のトランザクションログレコード１１２は、一実施態様における特殊なログレコードマーカ１６２を含む。本実施態様によると、シーケンス番号１６４は、Q_{get}またはQ_{put}のいずれかの操作を言及するメッセージ操作モード１６６とともに提供される。メッセージＩＤ１６８と、一組の操作フラグ１７０と、「アクティブ」、「ペンディング」「アボード」または「コミット」の状況を含むトランザクション状況１７２と、上述の参加者２ＰＣ投票１７４と、キューファイルにおけるメッセージのオンディスク位置を指すポインタ１７６が含まれる。

【００７０】次に図１１を参照すると、書込みまたはQ_{put}操作のフローチャートが示されている。本フローチャートでは、１８０で図示されるように開始した後、他のユーザがヘッド入力にアクセスできないようにブロックキューヘッドポインタ１８２が効果的に一覧のヘッドをロックする。その後、システムはキューヘッドポインタを増加し、トランザクション状況を「アクティブ読出し」に設定する。これは、ハンドシェークプロセスの開始を示す。

【００７１】１８６で図示されるように、システムはその後キューヘッドポインタのロックを解除し、その後１８８で図示されるように、メッセージをオンディスクキューファイルから読出す。次にQEMテーブルは、１９０で図示されるようにロックされ、ログレコードは、その後１９２で図示されるように書込まれ、QEMテーブルは、１９４で図示されるようにロックを解除される。QEMテーブルのロック解除ステップの出力は、メッセージ伝送がトランザクションのものであるかどうかを確認する決定ブロック１９６に当てはまる。トランザクションのものである場合、１９８で図示されるように、システムは二相「コミット」プロトコルを作動し、ハンドシェークを許容する。これによってQ_{put}または書込み操作が完了する。

【００７２】次に図１２を参照して、Q_{get}または読出し操作について説明する。２００で図示されるように開始されて、２０２で図示されるようにキューテイルポインタがロックされ、新規のQEM入力が、２０４で図示されるようにキューテイルポインタを増加させて作成される。その後、２０６で図示されるように、システムはQE

15

M入力制御情報を記入し、トランザクション状況を「アクティブ制御」に設定する。その後、２０８で図示されるように、キューテイルポインタはロックを解除され、QEMテーブルは、２１０で図示されるようにロックされる。

【００７３】次に、２１２で図示されるように、決定ブロック２１４で示されるセグメントの境界線を横切るブロックと共に、システムはオンディスクブロックを予約表から割り当てる。ブロックがセグメント境界線を横切る場合、２１６で図示されるように、システムはQEMTチェックポイントのディスクへの書込みを強制する。これは、メモリ内コピーをディスクに書込むことを指す。ブロック２０６は、QEMテーブルの状態のメモリ内コピーおよびこれによって得られるQEM入力を更新することが理解される。

【００７４】２１８で図示されるようにQEMTチェックポイントがディスクへの書込みを強制された後、当該システムは、メッセージデータをディスクに書込みQEMテーブルのロックを解除する。決定ブロック２２０は、メッセージがトランザクションのものであるかどうかを確定し、トランザクションのものである場合には、２２１で図示されるように二相コミットプロトコルを作動してハンドシェークを促進させる。書込みシーケンスの終了は２２２で図示される。ブロック２２０は、ハンドシェーキングプロトコルを動作している受け手末端を指す。

【００７５】次に図１３を参照すると、回復シーケンスが図示され、２３０で図示されるような開始後、キューテーブルポインタは２３２で図示されるようにロックされ、システムはその後、２３４で図示されるようにグローバルデータ構造を再格納する。このことは、システムの状態を全体的に初期化する。その後、２３６で図示されるように、システムは最新のQEMTについてキューファイルにおける各QEMTを走査する。これによって、通信割込み前に最新のチェックポイントが確立される。その後、２３８で図示されるように、システムは最新のQEMTを有するログレコードについてこのセグメントのログレコードを走査する。これは、セグメントのログレコードがQEMTにおいての入力によって照会されるメッセージに適用されることを意味する。

【００７６】決定ブロック２４０で図示されるように、システムは走査すべきログレコードがさらにあるかどうかを確認する。当該QEMTに関連するポインタに続いてQEMTは最新のログレコードを特定することが理解されよう。しかしながら、その後走査する必要のある次のログレコードが実際にある可能性がある。この場合、当該システムは、２４２で図示されるようにメッセージのトランザクション状況について参加者とコンタクトを取る。一例では、受信者はメッセージを受信したかどうかについて質問される。その後、システムは二相「コミット」プロトコルを呼出して、２４４で図示されるようにトランザクションを解決する。これは、ハンドシェーキングプロセスが2パスプロセスであることを示す。したがって、ある人が受信者から受信を返されるような状況であっても、この状況を利用して当該システムが停止した地点でハンドシェーキングプロセスを再開する。

16

【００７７】２４６で明らかなように、当該システムは、予約表の状況を更新し新規ファイルポインタ位置を決定する。したがって、現行のセグメント番号１３０および現行のセグメント１３６において次に利用可能なブロックによって新規ファイルのポインタ位置を決定しながら、セクション全体が走査され、予約表１４０の状況が更新される。

【００７８】２４８で図示されるように、システムは、２５０で図示されるように、回復が完了した地点で新規QEMTの状況をディスクに書き出す。

【００７９】上述したように、この発明の実施態様およびその変形例をここに記載したが、上述のものは単なる例として提示されたものであってこれに限定されるものではない。

【００８０】

【発明の効果】以上のように、この発明によれば、サーバ故障から迅速に回復できるようにメッセージおよびその状況を単一ディスク上の効率的な単一ファイルに保存し格納するメッセージキューイングシステムを提供することができる。

【図面の簡単な説明】

【図1】　メッセージが本店からその支店に流れる、当該システムを利用した典型的な銀行取引アプリケーションのブロック図である。

【図2】　データを一つのファイルに記録し、ログを別のファイルに記録し、前記データを非連続セクタに格納し、最新状況を再構築するためにログファイル全体を走査することを要求しながら、回復プロセスが当該システムの全メッセージの完全な状態を得るためにデータファイルおよびログファイルの両方を含む、２つのファイルシステムを表す線図である。

【図3】　単一ファイルを利用して、データおよびQEMTマッピングテーブルを格納することにより、ハードウェアを最小にし、データの回復に要する走査時間を短縮して紛失データを迅速に回復可能にする当該システムを表す線図である。

【図4】　単一書込み方向の巡回ファイルを示した、図３のファイル内にあるデータブロックの格納装置を表す線図である。

【図5】　QEMT制御ブロックを利用することによって有効データの位置および／または場所を容易に確認できることを示した、ファイル内の各種の周知の位置またはオフセットにおける可能なQEMT制御ブロックを表す線図である。

【図6】　ファイルをフォワード方向に書き込めるよう

17

に、メッセージデータブロックによって状態変化のログレコードが分散することを示す線図である。

【図7】　本QEMT構造を示し、タイムスタンプとして機能し、かつ当該システムを再格納するために必要とされる増分のチェックポイント情報を有した、QEMTシーケンス番号を含む表である。

【図8】　個別のメッセージ状況を再格納するための情報を提供する表である。

【図9】　巡回キューを実行する循環システムにおいてフォワード方向へのデータの流れを示す線図である。

【図10】　図6のログ入力によって、増分ログレコードに格納された情報を示す表である。

【図11】　キューからメッセージを取り出すための手順を示すフローチャートである。

18

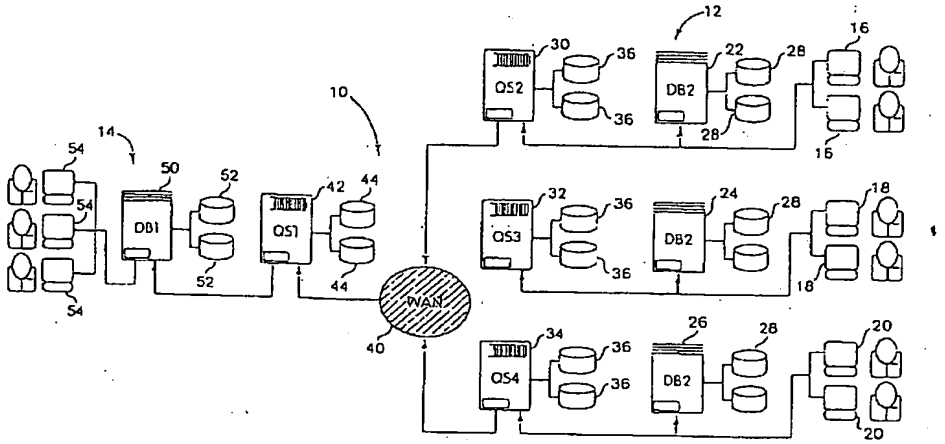
【図12】　メッセージをキューに書込むための手順を図示するフローチャートである。

【図13】　完全に再格納された状態をもたらす最新のQEMTを識別後にログレコードの次の読出しを行い、最新のQEMTが初期走査によって識別される回復プロセスを示すフローチャートである。

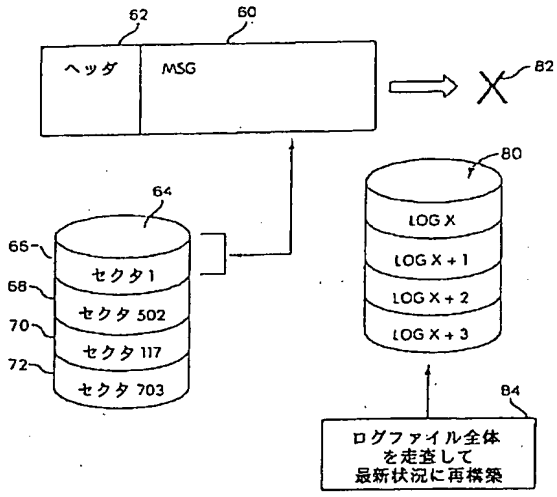
【符号の説明】

60　MSGデータ、62　MSGヘッダ情報、92　セクタ、94　セクタ、96　セクタ、98　セクタ、100　QEMT（制御情報）、102　MSGブロック、104　ログレコード。

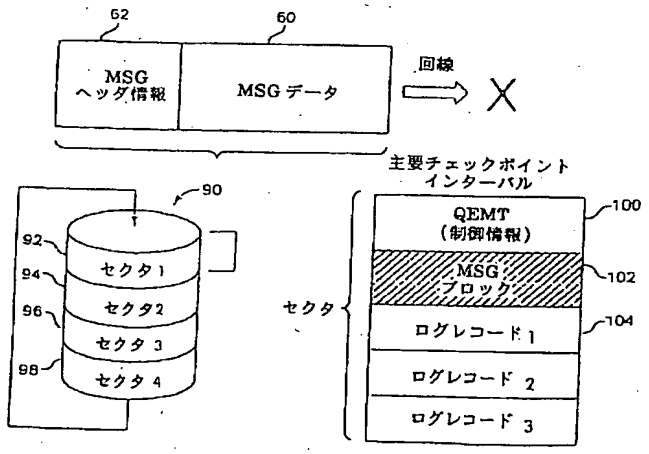
【図1】



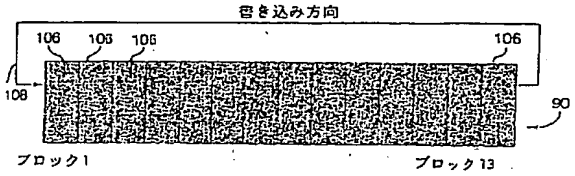
【図2】



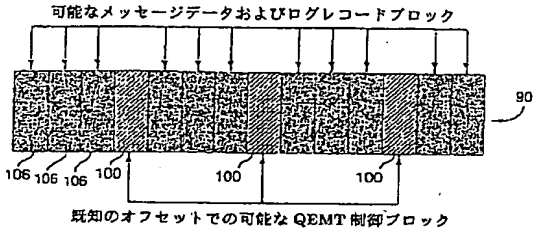
【図3】



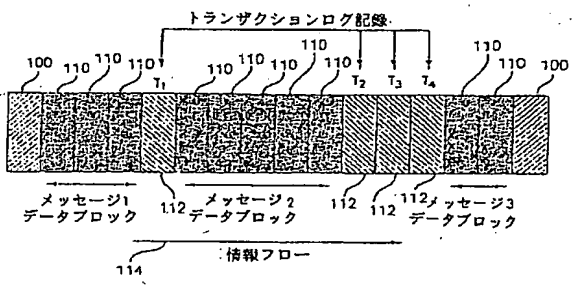
【 図4 】



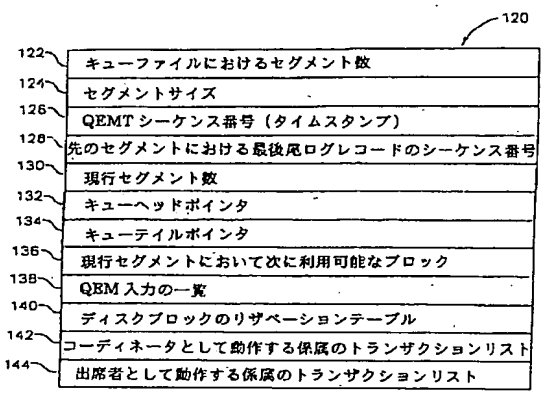
【 図5 】



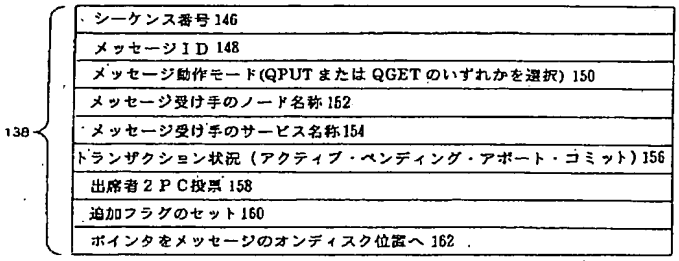
【 図6 】



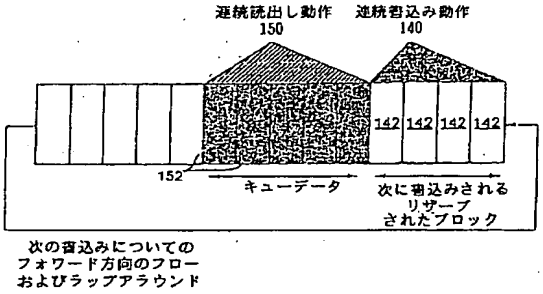
【 図7 】



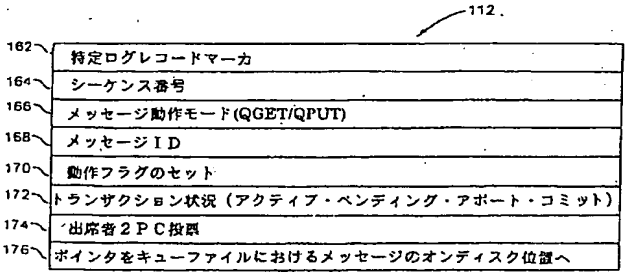
【 図8 】



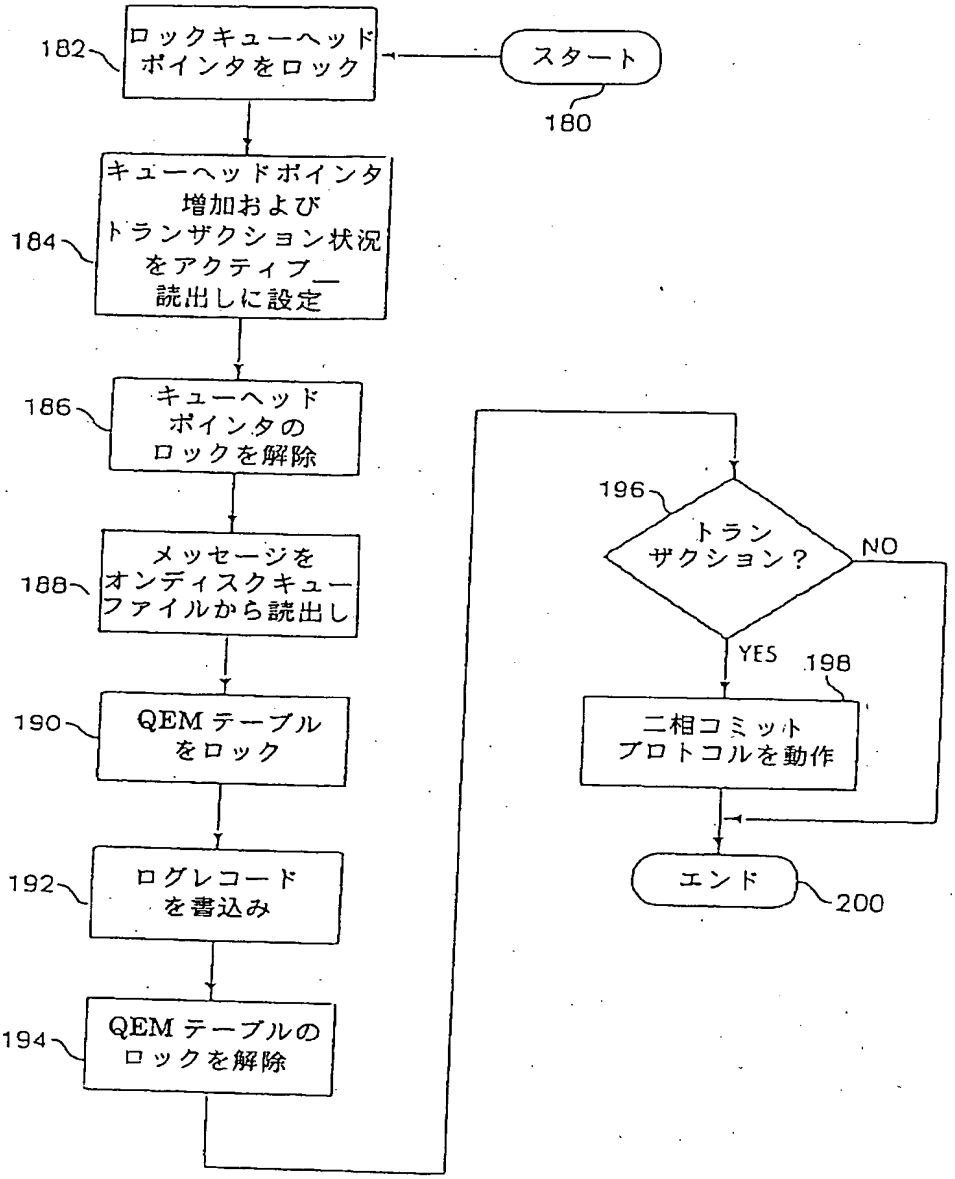
【 図9 】



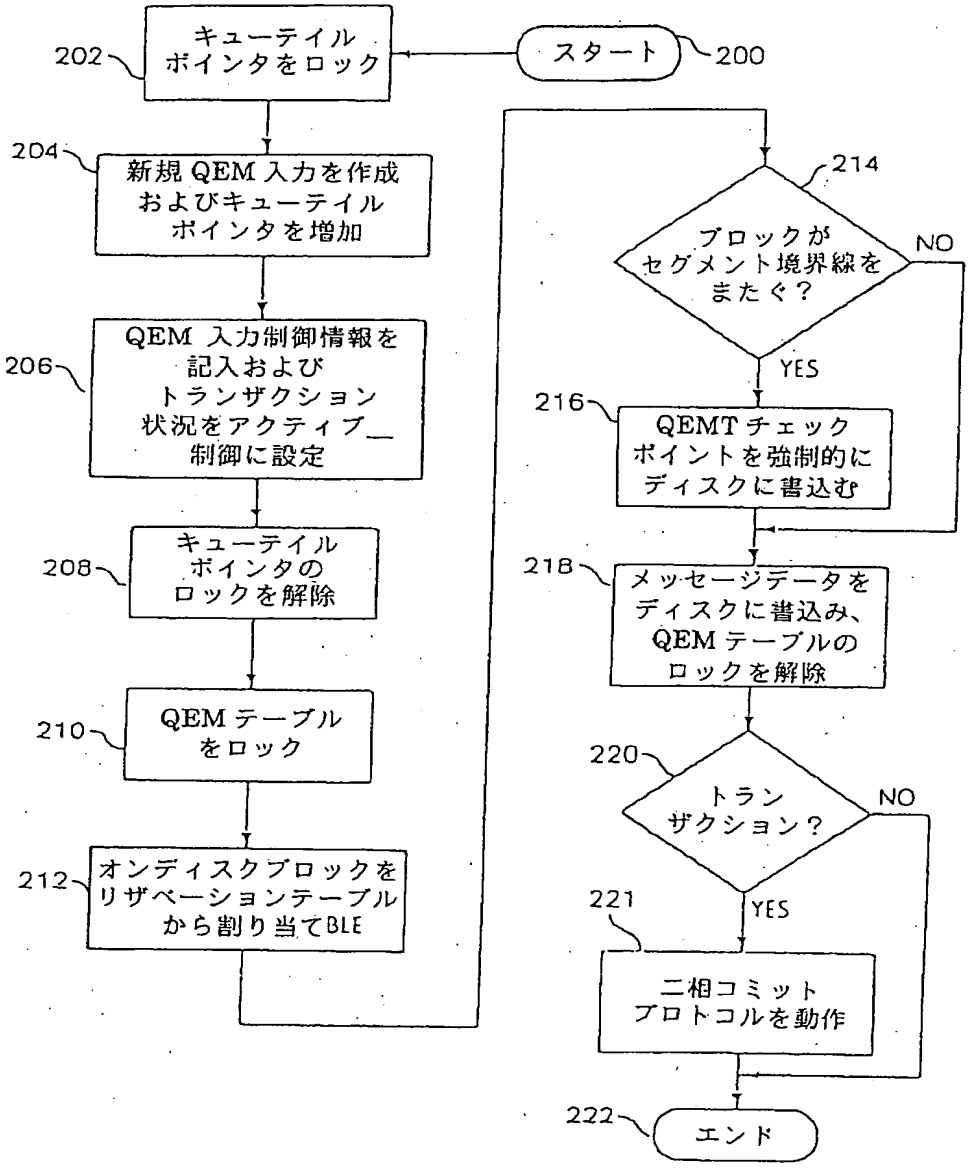
【 図10 】



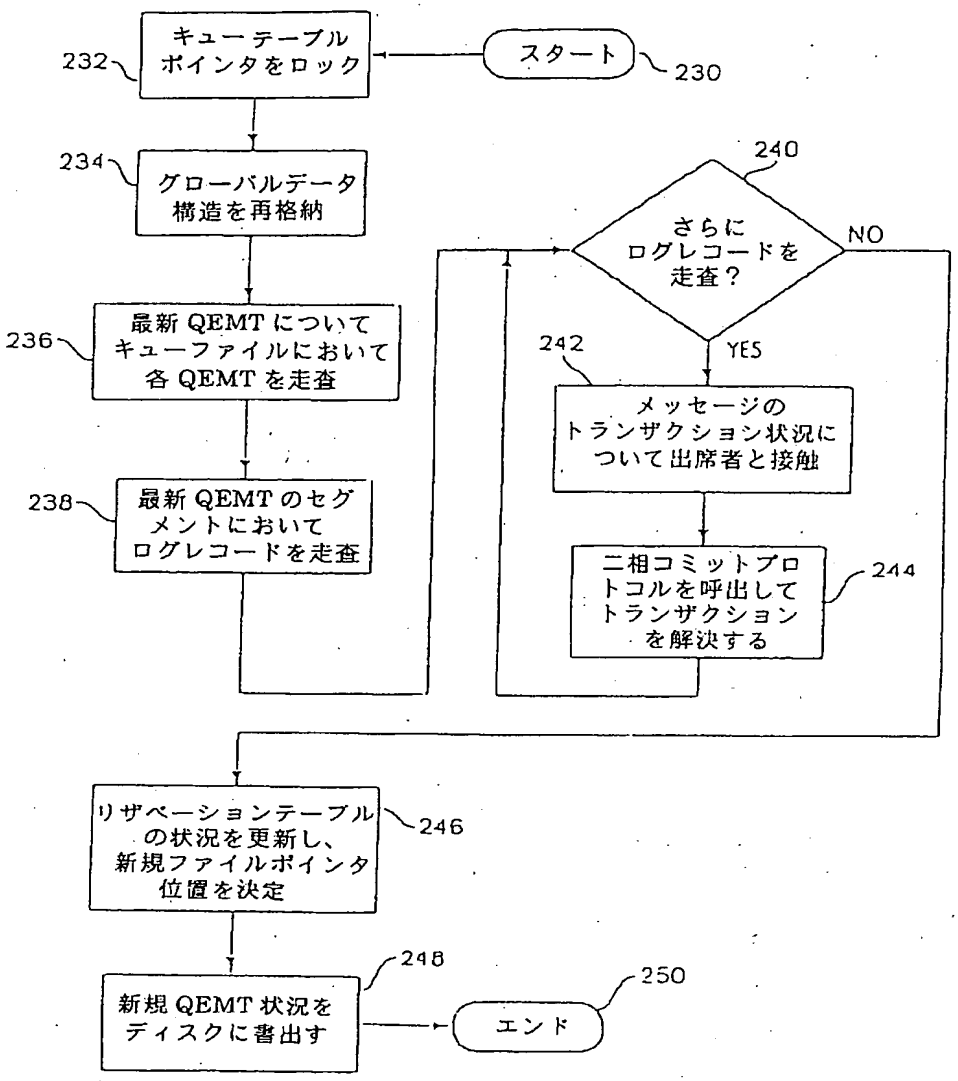
【図１１】


スタート
180
ロックキューヘッドポインタをロック
182
キューヘッドポインタ増加およびトランザクション状況をアクティブ＿読出しに設定
184
キューヘッドポインタのロックを解除
186
メッセージをオンディスクキューファイルから読出し
188
QEMテーブルをロック
190
ログレコードを書込み
192
QEMテーブルのロックを解除
194
トランザクション？
196
NO
YES
二相コミットプロトコルを動作
198
エンド
200

【図１２】


スタート
200
キューテイル
ポインタをロック
202
新規 QEM 入力を作成
およびキューテイル
ポインタを増加
204
QEM 入力制御情報を
記入および
トランザクション
状況をアクティブ
制御に設定
206
キューテイル
ポインタの
ロックを解除
208
QEM テーブル
をロック
210
オンディスクブロックを
リザベーションテーブル
から割り当てBLE
212
ブロックが
セグメント境界線を
またぐ？
214
NO
YES
QEMT チェック
ポイントを強制的に
ディスクに書込む
216
メッセージデータを
ディスクに書込み、
QEM テーブルの
ロックを解除
218
トラン
ザクション？
220
NO
YES
二相コミット
プロトコルを動作
221
エンド
222

【 図1.3 】



フロント ページの続き

(71)出願人　597067574
201 BROADWAY, CAMBRIDGE, MASSACHUSETTS 02139, U. S. A.

(72)発明者　デビッド・ダブリュ・エッチ・ワング
アメリカ合衆国、マサチューセッツ州、ボックスボロウ、グギンズ・レーン　162

(72)発明者　デレク・エル・シュベンケ
アメリカ合衆国、マサチューセッツ州、マールボロ、ライス・ストリート　95